

Programmed task based motion analysis of robotic systems equipped with flexible links and supports

Elżbieta Jarzębowska^{1*}, Krzysztof Augustynek², and Andrzej Urbaś²

¹Warsaw University of Technology, 00-665 Warsaw, Nowowiejska 24, Poland

²University of Bielsko-Biala, 43-309 Bielsko-Biala, Willowa 2, Poland

Abstract. The automatic computational procedure to derive dynamics equations of systems with programmed constraints modified to encompass compliant mechanical components in their structures is discussed in the paper. The dynamics analysis of the compliant manipulator model with a flexible link is presented as an example. The rigid finite element method is used in order to take into account the flexibility of the link. The formalism of joint coordinates and homogeneous transformations are used to describe the manipulator motion.

1 Introduction

Analysis of robot task based motions dedicated to work and services delivery is presented in the paper. The key contribution of the paper is in three aspects of the analysis. The first one is the possibility of constraints incorporation into the system dynamics, the second one is the special computational procedure for the constrained dynamics generation, which provides dynamic models, referred to as reference dynamics, satisfying all constraints upon them. The third contribution comparing to the results reported in the literature, see e.g. [1, 2] is in modeling flexibility of the system parts and supports. The constraints put on a system are referred to as programmed and they are imposed as control goals on system performance or service task requirements. The procedure of generating reference dynamics offers automated generation of equations and it was successfully developed and implemented to rigid system models [3, 4]. The advantage of this procedure is that it serves both reference and control oriented dynamics derivation and the final dynamics models are obtained in the reduced state form, i.e. constraint reaction forces are eliminated. The procedure is extended on flexible subsystems of lightweight and fast machine parts and servicing equipment, which are prone to vibration in some work regimes. Vibrations may significantly affect system performance and disable effective controller designs. Flexibility of links is modeled using the rigid finite element method, whose advantage relies in its ability of application of the rigid body approach to modeling flexible link elements. The novelty of the presented method is its ability to analyze any system reference

motion, including flexible link vibration, in the presence of kinematic task-based constraints. The results of this analysis can contribute to verification of a system behavior when it is subjected to given kinematic constraints, help to specify desired task-based constraints properly, e.g. put tighter velocity limits, to exclude some work regimes or to design controllers correctly. The possibility of reference motion analysis in these aspects enables simulation of various work regimes for system models and selection of required and safe task based motion parameters, accordingly. The theoretical development is illustrated by simulation studies of an example of a flexible link and support manipulator model, whose service tasks are predefined by the programmed constraints. Special interest of this study is to analyze flexible link vibrations when the manipulator performs the tasks.

2 Mathematical model of a manipulator

Figure 1 presents a compliant supported manipulator model. It is assumed, that the manipulator is driven by means of a driving torque $\mathbf{t}_{dr}^{(2)}$. The manipulator third link is flexible. The rigid finite element method is used to discretize this link.

The following joint coordinates are used to describe motion of the manipulator:

$$\mathbf{q} = (q_i)_{i=1, \dots, n_{dof}} = \left[\tilde{\mathbf{q}}^{(1)T} \mid \tilde{\mathbf{q}}^{(2)T} \mid \tilde{\mathbf{q}}^{(3)T} \mid \tilde{\mathbf{q}}^{(4)T} \right]^T, \quad (1)$$

where: $\tilde{\mathbf{q}}^{(1)} = \left[z^{(1)} \right]$, $\tilde{\mathbf{q}}^{(2)} = \left[\psi^{(2)} \right]$,

* Corresponding author: elajarz@meil.pw.edu.pl

$$\tilde{\mathbf{q}}^{(3)} = \begin{bmatrix} \tilde{\mathbf{q}}^{(3,0)^T} & \dots & \tilde{\mathbf{q}}^{(3,r)^T} & \dots & \tilde{\mathbf{q}}^{(3,n_{jfc}^{(3)})^T} \end{bmatrix}^T,$$

$$\tilde{\mathbf{q}}^{(3,0)} = [\psi^{(3,0)}], \quad \tilde{\mathbf{q}}^{(3,r)} = [\psi^{(3,r)} \quad \theta^{(3,r)} \quad \phi^{(3,r)}]^T,$$

$$\tilde{\mathbf{q}}^{(4)} = [z^{(4)}].$$

Vector \mathbf{q} contains independent and dependent coordinates:

$$\mathbf{q} = (q_i)_{i \in i_c \cup i_{d_c}} \in \{\mathbf{q}_{i_c}, \mathbf{q}_{d_c}\}, \quad (2)$$

where: $i_c \in \{1, 2, 4, \dots, n_{dof-1}\}$ and $i_{d_c} \in \{3, \dots, n_{dof}\}$ are independent and dependent coordinates indexes.

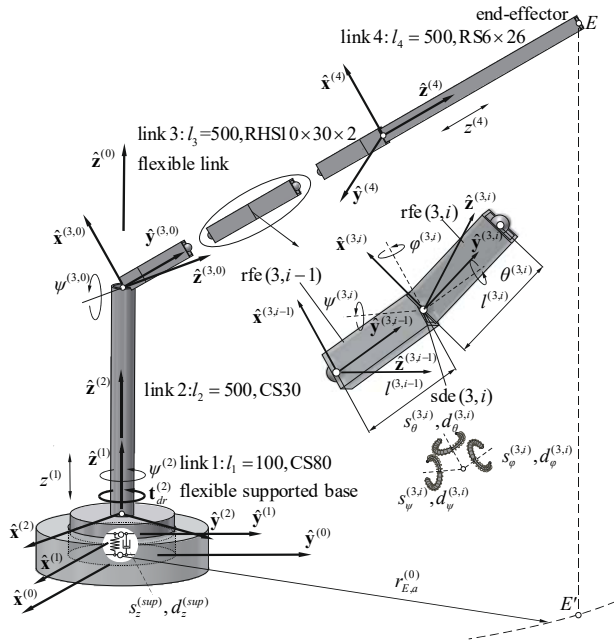


Figure 1. Model of a manipulator.

The generalized programmed motions equations (GPME) with first order programmed constraints underlie the derivation procedure and they have the form [3, 4]:

$$\frac{\partial R_i}{\partial \dot{q}_i} \Big|_{i \in i_c} + \sum_{j \in i_{d_c}} \frac{\partial R_i}{\partial \dot{q}_j} \frac{\partial \dot{q}_j}{\partial \dot{q}_i} = 0, \quad (3)$$

where:

$$R_i = \dot{E}_k - 2 \sum_{i=1}^{n_{dof}} \frac{\partial E_k}{\partial \dot{q}_i} \dot{q}_i + \sum_{i=1}^{n_{dof}} \frac{\partial E_p}{\partial \dot{q}_i} \dot{q}_i + \sum_{i=1}^{n_{dof}} \frac{\partial R_s}{\partial \dot{q}_i} \dot{q}_i - \sum_{i=1}^{n_{dof}} t_i \dot{q}_i,$$

$$E_k = \sum_{l=1}^{n_l} E_k^{(l)}, \quad E_k^{(l)} = \frac{1}{2} \text{tr} \left\{ \dot{\mathbf{T}}^{(l)} \mathbf{H}^{(l)} \left(\dot{\mathbf{T}}^{(l)} \right)^T \right\},$$

$$E_p = E_{p,g} + E_{p,sup} + E_{p,f_i}, \quad E_{p,g} = \sum_{l=1}^{n_l} E_{p,g}^{(l)},$$

$$E_{p,g}^{(l)} = m^{(l)} \mathbf{g} \mathbf{J}_3 \mathbf{T}^{(l)} \mathbf{r}_{C^{(l)}}, \quad \mathbf{J} = \begin{bmatrix} \mathbf{J}_1 \\ \mathbf{J}_2 \\ \mathbf{J}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$$E_{p,sup} = \frac{1}{2} s_z^{(sup)} \left(z^{(1)} \right)^2,$$

$$E_{p,f_i} = \frac{1}{2} \sum_{s=1}^{n_s^{(i)}} \left(\tilde{\mathbf{q}}^{(l,s)} \right)^T \mathbf{S}^{(l,s)} \tilde{\mathbf{q}}^{(l,s)}, \quad \mathbf{S}^{(l,s)} = \text{diag} \left\{ s_\psi^{(l,s)}, s_\theta^{(l,s)}, s_\phi^{(l,s)} \right\},$$

$$R_s = R_{sup} + R_{f_i}, \quad R_{sup} = \frac{1}{2} d_z^{(sup)} \left(z^{(1)} \right)^2,$$

$$R_{f_i} = \frac{1}{2} \sum_{s=1}^{n_s^{(i)}} \left(\dot{\tilde{\mathbf{q}}}^{(l,s)} \right)^T \mathbf{D}^{(l,s)} \dot{\tilde{\mathbf{q}}}^{(l,s)}, \quad \mathbf{D}^{(l,s)} = \text{diag} \left\{ d_\psi^{(l,s)}, d_\theta^{(l,s)}, d_\phi^{(l,s)} \right\},$$

$$\mathbf{t} = \left[0 \quad t_{dr}^{(2)} \quad \mathbf{0} \right]^T.$$

It is assumed that the end-effector moves along an elliptical trajectory designed in a plane parallel to the $x^{(0)}y^{(0)}$ plane and its $z_E^{(0)}$ coordinate has to change according to assumed time-dependent function $z_{E,a}^{(0)}(t)$.

Thus, these programmed constraints can be written in the form:

$$\Phi_1 \equiv 0 \Rightarrow \left(\frac{x_E^{(0)}}{a_{E,a}^{(0)}} \right)^2 + \left(\frac{y_E^{(0)}}{b_{E,a}^{(0)}} \right)^2 - 1 = 0 \quad (4.1)$$

$$\Phi_2 \equiv 0 \Rightarrow z_E^{(0)} - z_{E,a}^{(0)}(t) = 0 \quad (4.2)$$

where: $x_E^{(0)} = \mathbf{J}_1 \mathbf{T}^{(4)} \mathbf{r}_E^{(4)}$, $y_E^{(0)} = \mathbf{J}_2 \mathbf{T}^{(4)} \mathbf{r}_E^{(4)}$, $z_E^{(0)} = \mathbf{J}_3 \mathbf{T}^{(4)} \mathbf{r}_E^{(4)}$,

According to the GPME algorithm [3], the constraint equations are differentiated to obtain

$$\dot{\Phi}_1 \equiv 0 \Rightarrow \mathbf{u} \dot{\mathbf{q}} = \mathbf{0}, \quad (5.1)$$

$$\dot{\Phi}_2 \equiv 0 \Rightarrow \mathbf{C}_3 \dot{\mathbf{q}} - \dot{z}_{E,a}^{(0)}(t) = 0, \quad (5.2)$$

$$\ddot{\Phi}_1 \equiv 0 \Rightarrow \mathbf{u} \ddot{\mathbf{q}} + \mathbf{v} = 0, \quad (5.3)$$

$$\ddot{\Phi}_2 \equiv 0 \Rightarrow \mathbf{C}_3 \ddot{\mathbf{q}} + d_3 - \ddot{z}_{E,a}^{(0)}(t) = 0, \quad (5.4)$$

with:

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \\ \mathbf{C}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1 \\ (c_{ij})_{i=1,2,3} \\ \mathbf{C}_3 \end{bmatrix} = \mathbf{J} \left[\mathbf{T}_1^{(4)} \mathbf{r}_E^{(4)} \quad \dots \quad \mathbf{T}_4^{(4)} \mathbf{r}_E^{(4)} \right],$$

$$\mathbf{d} = (d_i)_{i=1, \dots, 3} = \mathbf{J} \left(\sum_{i=1}^{n_{dof}} \sum_{j=1}^{n_{dof}} \mathbf{T}_{ij}^{(4)} \dot{q}_i \dot{q}_j \right) \mathbf{r}_E^{(4)},$$

$$\mathbf{u} = (u_j)_{j=1, \dots, 4} = \frac{1}{(a_{E,a}^{(0)})^2} \mathbf{J}_1 \mathbf{T}^{(4)} \mathbf{r}_E^{(4)} \mathbf{C}_1 + \frac{1}{(b_{E,a}^{(0)})^2} \mathbf{J}_2 \mathbf{T}^{(4)} \mathbf{r}_E^{(4)} \mathbf{C}_2,$$

$$\mathbf{v} = \frac{1}{(a_{E,a}^{(0)})^2} \left((\mathbf{C}_1 \dot{\mathbf{q}})^2 + \mathbf{J}_1 \mathbf{T}^{(4)} \mathbf{r}_E^{(4)} d_1 \right) +$$

$$+ \frac{1}{(b_{E,a}^{(0)})^2} \left((\mathbf{C}_2 \dot{\mathbf{q}})^2 + \mathbf{J}_2 \mathbf{T}^{(4)} \mathbf{r}_E^{(4)} d_2 \right),$$

$a_{E,a}^{(0)}$, $b_{E,a}^{(0)}$ are the elliptical trajectory semi-major and semi-minor axes, respectively.

Vector of dependent velocities are as follows:

$$\dot{\mathbf{q}}_{d_c} = -\mathbf{K}_{d_c}^{-1} \mathbf{K}_{i_c} \dot{\mathbf{q}}_{i_c}, \quad (6)$$

where:

$$\mathbf{K}_{d_c} = \begin{bmatrix} u_2 & u_4 \\ c_{12} & c_{14} \end{bmatrix}, \quad \mathbf{K}_{i_c} = \begin{bmatrix} u_1 & u_3 \\ c_{11} & c_{13} \end{bmatrix}.$$

The GPME equations supplemented by the equations of programmed constraints take the form:

$$\begin{bmatrix} \mathbf{M}_i \Big|_{i \in i_c} + \sum_{j \in i_{d_c}} \mathbf{M}_j \frac{\partial \dot{q}_j}{\partial \dot{q}_i} \\ \mathbf{u} \\ \mathbf{C}_3 \end{bmatrix} \ddot{\mathbf{q}} = \begin{bmatrix} \mathbf{h}_i + \mathbf{Q}_i + \sum_{j \in i_c \cup i_{d_c}} \dot{q}_j \frac{\partial Q_j}{\partial \dot{q}_i} + \dots \\ \dots \sum_{k \in i_{d_c}} \left(h_k + Q_k + \sum_{j \in i_c \cup i_{d_c}} \dot{q}_j \frac{\partial Q_j}{\partial \dot{q}_k} \right) \frac{\partial \dot{q}_k}{\partial \dot{q}_i} \\ -v \\ -d_3 + \ddot{z}_{E,a}^{(0)}(t) \end{bmatrix}, \quad (7)$$

where:

$$\mathbf{M} = \sum_{l=1}^{n_l} \mathbf{M}^{(l)}, \quad \mathbf{M}^{(l)} = (m_{ij}^{(l)})_{i,j=1,\dots,n_{dof}^{(l)}},$$

$$m_{ij}^{(l)} = \text{tr} \left\{ \mathbf{T}_i^{(l)} \mathbf{H}^{(l)} \left(\mathbf{T}_j^{(l)} \right)^T \right\},$$

$$\mathbf{h} = \sum_{l=1}^{n_l} \mathbf{h}^{(l)}, \quad \mathbf{h}^{(l)} = (h_i^{(l)})_{i=1,\dots,n_{dof}^{(l)}},$$

$$h_i^{(l)} = \sum_{m=1}^{n_{dof}^{(l)}} \sum_{n=1}^{n_{dof}^{(l)}} \text{tr} \left\{ \mathbf{T}_m^{(l)} \mathbf{H}^{(l)} \left(\mathbf{T}_{m,n}^{(l)} \right)^T \right\} \dot{q}_m^{(l)} \dot{q}_n^{(l)} + 2 \sum_{m=1}^{n_{dof}^{(l)}} \sum_{n=1}^{n_{dof}^{(l)}} \text{tr} \left\{ \mathbf{T}_m^{(l)} \mathbf{H}^{(l)} \left(\mathbf{T}_{i,n}^{(l)} \right)^T \right\} \dot{q}_m^{(l)} \dot{q}_n^{(l)},$$

$$\mathbf{T}_i^{(l)} = \frac{\partial \mathbf{T}^{(l)}}{\partial q_i^{(l)}}, \quad \mathbf{T}_{i,j}^{(l)} = \frac{\partial^2 \mathbf{T}^{(l)}}{\partial q_i^{(l)} \partial q_j^{(l)}},$$

$$\mathbf{Q} = -(\mathbf{g} + \mathbf{d} + \mathbf{f}) + \mathbf{t},$$

$$\mathbf{g} = \sum_{l=1}^{n_l} \mathbf{g}^{(l)}, \quad \mathbf{g}^{(l)} = (g_i^{(l)})_{i=1,\dots,n_{dof}^{(l)}}, \quad g_i^{(l)} = m^{(l)} g \mathbf{J}_1 \mathbf{T}_i^{(l)} \mathbf{r}_{C^{(l)}}^{(l)},$$

$$\mathbf{f} = [f^{(1)} \quad 0 \quad \mathbf{f}^{(3)} \quad 0]^T,$$

$$f^{(1)} = s_z^{(sup)} z^{(1)}, \quad \mathbf{f}^{(3)} = \begin{bmatrix} 0 & \mathbf{S}^{(3,1)} \tilde{\mathbf{q}}^{(3,1)} & \dots & \mathbf{S}^{(3,n_{sdc}^{(3)})} \tilde{\mathbf{q}}^{(3,n_{sdc}^{(3)})} \end{bmatrix},$$

$$\mathbf{d} = [d^{(1)} \quad 0 \quad \mathbf{d}^{(3)} \quad 0]^T,$$

$$d^{(1)} = d_z^{(sup)} z^{(1)}, \quad \mathbf{d}^{(3)} = \begin{bmatrix} 0 & \mathbf{D}^{(3,1)} \dot{\tilde{\mathbf{q}}}^{(3,1)} & \dots & \mathbf{D}^{(3,n_{sdc}^{(3)})} \dot{\tilde{\mathbf{q}}}^{(3,n_{sdc}^{(3)})} \end{bmatrix},$$

\mathbf{H} - pseudo-inertia matrix, g - the acceleration of gravity.

3 Simulation study – programmed motion of the crane end-effector

Simulation studies are carried out for two cases: the manipulator with rigid links and flexible support and the manipulator with the flexible link and flexible support. The following parameters are used in simulation: $s_z^{(sup)} = 10^4 \text{ Nm}^{-1}$, $d_z^{(sup)} = 2.5 \text{ Nsm}^{-1}$, $a_{E,a}^{(0)} = 0.375 \text{ m}$,

$b_{E,a}^{(0)} = 0.750 \text{ m}$. It is assumed that the end-effector $z_E^{(0)}$ coordinate changes according to the following function:

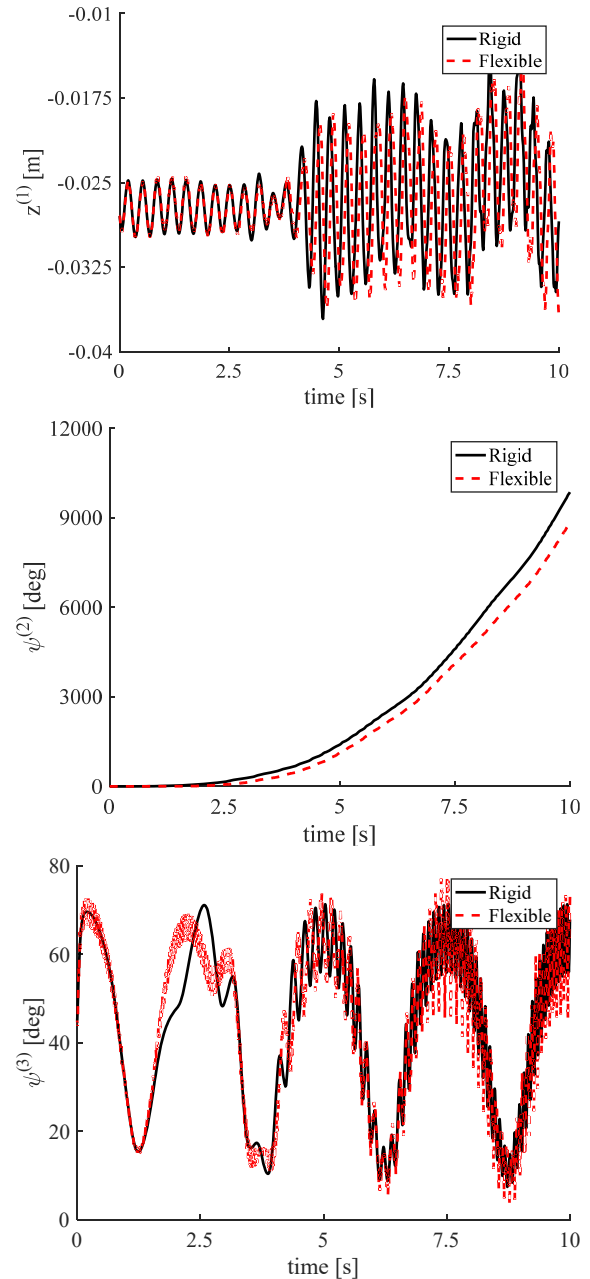
$$z_{E,a}^{(0)}(t) = z_0 + A \cos\left(\frac{2\pi}{T}t\right), \quad (8)$$

where z_0 is the initial position of the end-effector, A is the amplitude and T is a time period of oscillations. It is assumed that $A = 0.5 \text{ m}$ and $T = 2.5 \text{ s}$. The column (2) is driven by the driving torque defined by:

$$t_{dr}^{(1)}(t) = \begin{cases} 10t_{dr,0}^{(1)}\left(\frac{t}{t_0}\right)^3 - 15t_{dr,0}^{(1)}\left(\frac{t}{t_0}\right)^4 + 6t_{dr,0}^{(1)}\left(\frac{t}{t_0}\right)^5, & t < t_0, \\ t_{dr,0}^{(1)}, & t \geq t_0. \end{cases} \quad (9)$$

It is assumed $t_0 = 5 \text{ s}$ and $t_{dr,0}^{(1)} = 1 \text{ Nm}$.

The dynamic equations of motion are integrated using 4th order Runge-Kutta scheme with using the Baumgarte stabilization method for the programmed constraint equations. Figure 2 shows time courses of the joint coordinates obtained from simulations.



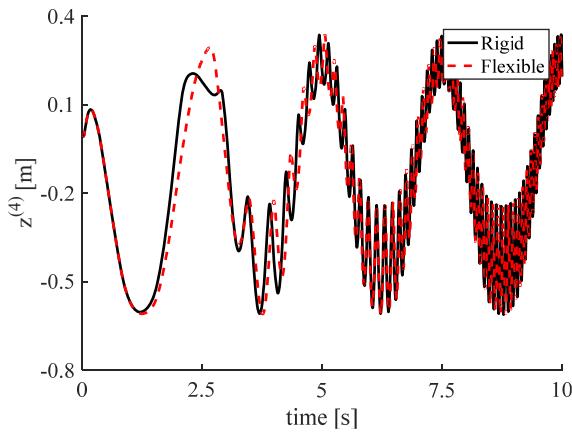


Figure 2. Time course of the joint coordinates

It can be observed that the link flexibility has significant influence on motion of the manipulator. The results show that the programmed motion of the manipulator is compensated by the appropriate combination of the rotation angle of the arm 3 and the displacement of the slider 4. When the flexibility is taken into account additional vibrations with small amplitude and large frequency are induced at initial phase of the motion. After 5 s, the amplitude of vibrations of arm 3 is larger than those obtained for the rigid body model.

Figure 3 presents the course of programmed constraints.

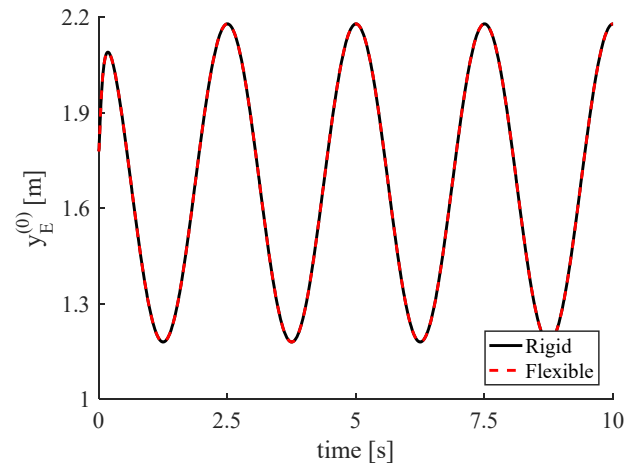
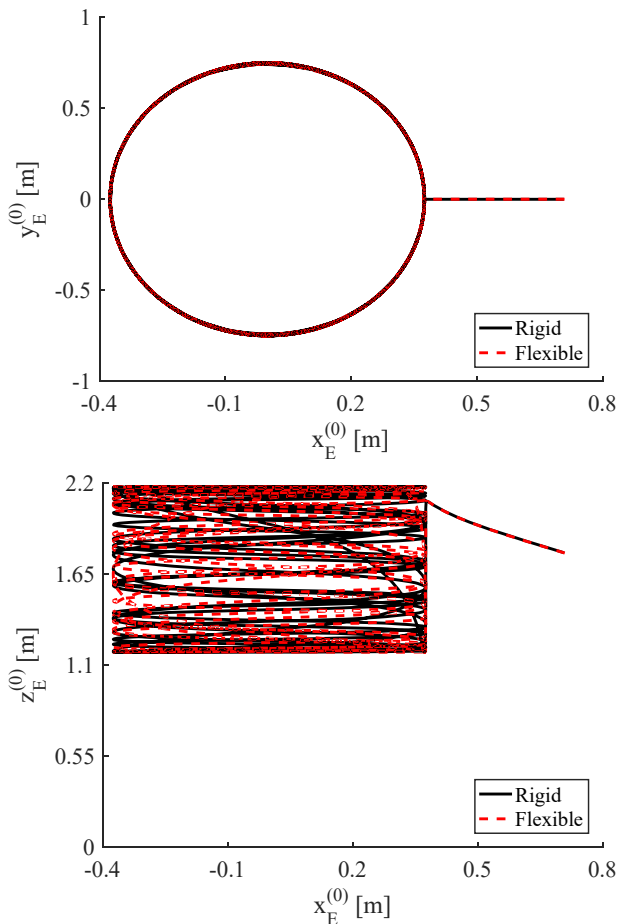


Figure 3. The end-effector trajectory and the time course of $z_E^{(0)}$ coordinate

It can be observed a good agreement of the obtained courses with the assumed programmed constraints. At the initial phase of motion the end-effector is located outside the assumed trajectory and after some period of time this trajectory is achieved.

4 Conclusions

The computational reference model of a crane equipped with the flexible link and support subjected to first order programmed constraints is presented. It is generated by the automatic computational algorithm designed using the formalism of joint coordinates and homogeneous transformations. The algorithm enables constraints incorporation into the system dynamics and modeling flexibility of the system parts and supports. The presented approach can be generalized for systems with higher order constraints.

References

1. A.A. Ata, H. Johar, International Journal of Advanced Robotic Systems, **1**, 2 (2004)
2. A.A. Ataa, W.F. Faresb, M.Y. Sa'adehc, Procedia Engineering, **41** (2012)
3. E. Jarzębowska, K. Augustynek, A. Urbaś, Concurrency and Computation: Practice and Experience (2018), doi:10.1002/cpe.4452. 25
4. E. Jarzębowska, K. Augustynek, A. Urbaś, Proc. of ASME Design Eng. Techn. Conf., No UNSP V006T10A016 (2017)